

aborting response preparation to a client request if the client-to-server channel is determined to be no longer established.

2. The method of claim 1, wherein a state of the server-to-client channel is inferred according to whether the client-to-server channel is still established; and wherein the response preparation is aborted if the server-to-client channel is inferred to be closed.

3. The method of claim 1, wherein the server includes a read buffer; wherein the client request is read from the read buffer; and wherein the read buffer is then probed to determine whether the client-to-server channel is still established.

4. The method of claim 1, wherein the server maintains local information about the state of the client-to-server channel; wherein a specific state of the client-to-server channel is determined by examining the local information; and wherein the response preparation is aborted if the local information indicates that the client-to-server channel is in the specific state.

5. The method of claim 4, wherein the client-to-server channel is determined to be no longer established if the local information indicates that the client-to-server channel is in a "CLOSE_WAIT" state.

6. The method of claim 1, wherein the state of the client-to-server channel is determined by polling the local information while a response to the client request is being prepared, whereby response preparation can be aborted while a request is being prepared.

7. The method of claim 1, further comprising generating an interrupt when the client-to-server channel is determined to be no longer established, wherein a response to the client request is processed until the interrupt is generated.

8. A network server comprising:
a processing unit;
a network interface card; and
computer memory programmed to cause the processing unit to examine local server information to determine whether a client-to-server channel is still established; and abort response preparation if the client-to-server channel is determined to be no longer established.

9. The server of claim 8, wherein a state of a server-to-client channel is inferred according to whether the client-to-server channel is still established; and wherein the response preparation is aborted if the server-to-client channel is inferred to be closed.

10. The server of claim 9, further comprising a read buffer; wherein a client request is read from the read buffer; and wherein the read buffer is probed to determine whether the client-to-server channel is still established

11. The server of claim 8, wherein the memory includes local information about a state of the client-to-server channel; wherein a state of the client-to-server channel is determined by examining the local information; and wherein the response preparation is aborted if the local information indicates that the client-to-server channel is in the specific state.

a
Sub
C1

12. The server of claim 11, wherein the client-to-server channel is determined to be no longer established if the local information indicates that the client-to-server channel is in a "CLOSE_WAIT" state.

13. The server of claim 8, wherein a state of the client-to-server channel is determined by polling the local information while a response to the client request is being prepared.

14. The server of claim 8, wherein the memory is programmed with a routine for commanding the processing unit to generate an interrupt when the client-to-server channel is determined to be no longer established, and wherein a response to a client request is processed until the interrupt is generated.

15. A network server comprising:

a processing unit;

first means for maintaining a queue of connections based on connection requests, each network connection including a client-to-server channel and a server-to-client channel;

second means for accepting connections from the queue;

third means for examining local server information to determine whether the client-to-server channel of a given connection from the queue is still established; and

fourth means for aborting response preparation if it is determined that the client-to-server channel of the given connection is no longer established.

16. An article for a network server including a processing unit and a network interface card, the article comprising:

computer memory; and

Sub
C1

a server program encoded in the computer memory, the server program commanding the processing unit to accept network connections, each connection having a client-to-server channel and a server-to-client channel; examine local server information to determine whether the client-to-server channel of a given connection from the queue is still established; and abort response preparation if the client-to-server channel of the given connection is determined to be no longer established.

17. The article of claim 16, wherein a state of the server-to-client channel of the given connection is inferred according to whether the corresponding client-to-server channel is still established.

18. The article of claim 16, wherein the memory is further encoded with local information about a state of the given connection; wherein the state of the given connection is determined by examining the local information; and wherein response preparation is aborted if the local information indicates that the client-to-server channel of the given connection is in the specific state.

19. The article of claim 16, wherein a state of the client-to-server channel of the given connection is determined by polling the local information, the local information being polled while a response to a client request is being prepared.

20. The article of claim 16, wherein the memory is further encoded with a routine for commanding the processing unit to generate an interrupt when the client-to-server channel of the given connection is determined to be no longer established, and wherein a response to a client request is processed until the interrupt is generated.

21. A computer program for a processing unit, the program comprising instructions for commanding a processing unit to maintain a queue of network connections based on connection requests, the program further comprising instructions for commanding the processing unit to accept connections from the queue; examine local server information to determine whether a client-to-server channel of a given connection from the queue is still established; process a client request associated with the given connection if the client-to-to-server channel of the given connection is still established; and abort response preparation for the associated client request if the client-to-server channel of the given connection is no longer established.

Sub
C 2